Ozone and the Role It Plays in Human Life

Brandelyn Nie

2022-11-21

Abstract

Analysis on this data was done with the motivation to explore data related to environmental issues related to ozone. "Ozone is a bluish toxic gas in the earth's stratosphere. It irritates lung tissue, causes inflammation, and is considered to be one of the most harmful compounds in air pollution" (Hitti). In this report we modeled how ozone concentration changes over time using time series analysis techniques and created a forecast for the year after the data used to create the model. The time series techniques used are exploratory data analysis, transformation to stationary data, model identification, model diagnostics, and forecasting. In the report, we conclude that ozone has a seasonal pattern and has been slowly decreasing over time. Research is done on what environmental factors might explain the seasonality and trend, notably increased pollutants and hot weather having interesting correlations with ozone levels. The results of this report can be used to determine what changes to environmental policy may be needed to maintain a healthy planet for us to live on.

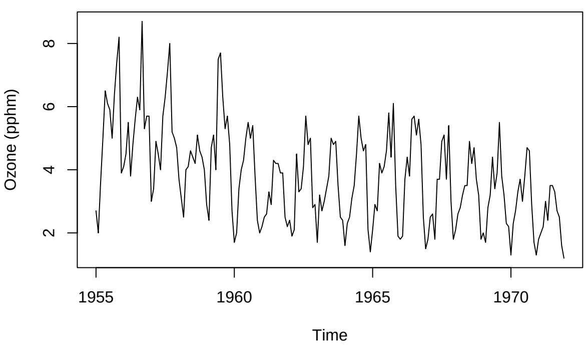
Introduction

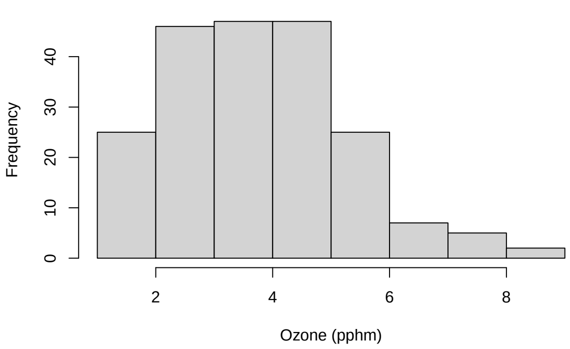
Changes in ozone concentration is an important topic to consider as ozone in the atmosphere protects humans from harmful UV rays, and there is also ozone in the troposphere that causes harm to humans. Modeling ozone concentration over time is helpful to find potential patterns, and then figure out what might have caused these problems. In this project, consider a data set about the ozone concentration in parts per hundred million in downtown Los Angeles from the years 1955 to 1972. This data set is important because this city has a lot of pollution production, and connections can be made about where certain seasonal patterns and trends come from with the time series model. To forecast and see the validity of the model, a training set (U_t) and test set were created, with the test set being the last 12 months removed. The training set was used to build the model, and the test set to validate how well it forecasted. A forecast on the change of ozone over time is important to see as high levels of ozone is detrimental to human health, and can help people that may be more susceptible to these changes, such as those with weaker respiratory systems to stay indoors.

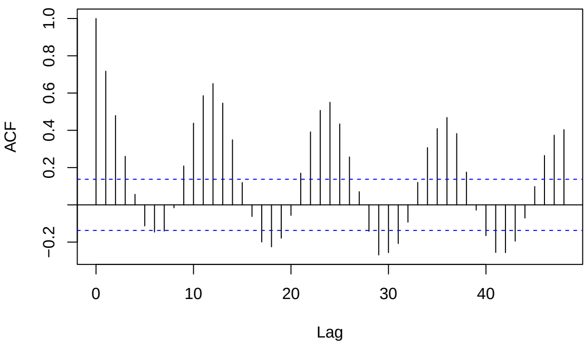
In this report, the data is from Hipel and McLeod (1994), and the report and graphs were compiled with the R programming language and RMarkdown.

Exploratory Data Analysis

First plot the time series and see if it is a stationary series. For a stationary series, there should be no trend, no seasonality, constant variance, and no apparent sharp changes in behavior.


Ozone concentration, downtown L. A., 1955−1972


Ozone concentration, downtown L. A., 1955−1972


ACF of Ozone concentration, downtown L. A.

Looking at the time series plot, there is a decreasing trend. Seasonality is present as well, since there is a notable pattern increasing and decreasing each year. Plotting the ACF further shows that seasonality is present as the ACFs remain large and periodic. There is no apparent sharp change in behavior, as there is a fairly consistent rise and fall with the peaks throughout the years, and a slowly decreasing trend.

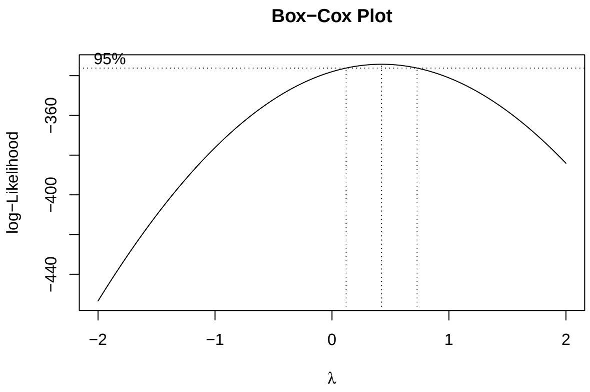
There are some really tall and some short peaks in the time series plot and further analysis of the histogram shows that the data is right skewed. These two observations of the data show that the data is non-constant.

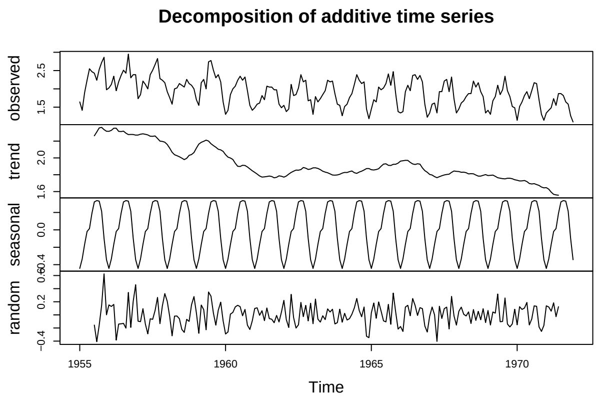
Transformation to Stationary Series

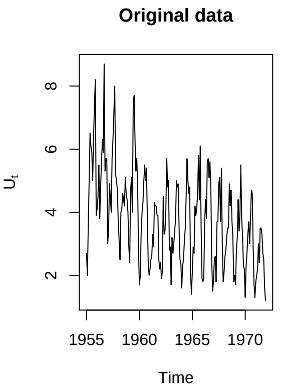
As mentioned, there is non-constant variance in the time series, so proceed to transform the data to make it normally distributed. To do this, use the Box-Cox Method, which will compute log-likelihoods and find the optimal lambda to take the data to the power of to make it normally distributed. From the plot, the lambda to use should fall between the dotted interval, with the most optimal value being the vertical line in the middle of the interval. For easier interpretation, select $\lambda = \frac{1}{2}$, which corresponds to a square root transformation of the data.

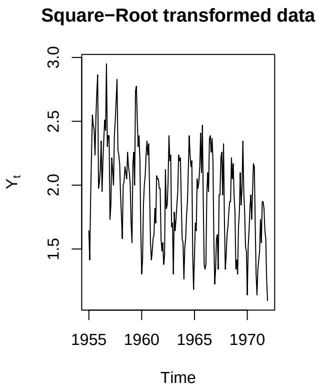
Look at the original (U_t) and transformed data $(Y_t = U_t^{\frac{1}{2}})$ side by side to see that the variance does stabilize with a square root transformation, since the tall peaks have been shortened. This is further supported by plotting the histogram of the transformed data, which is now normally distributed.

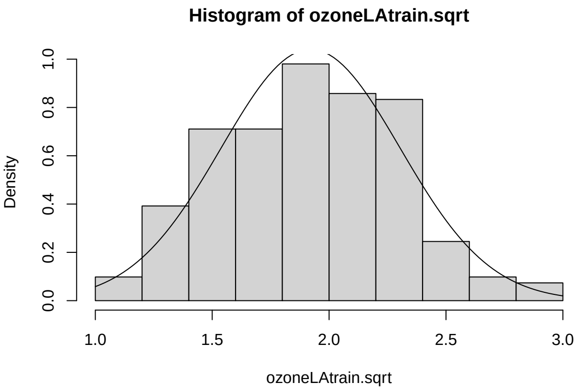
The decomposition of the transformed time series shows seasonality and trend, so differencing is needed to make the data stationary.







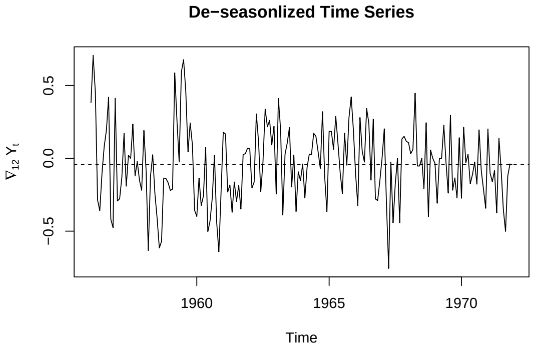




De-season and De-trend

The data is still seasonal and has a trend, so try a few differences at different lags. Compared to the original data, taking the difference at lag 12 is the only option that decreases the variance. Plot the data differenced at lag 12, and find that it is de-trended and de-seasoned, and so $\nabla_{12}Y_t$ is a stationary time series. This is satisfactory to continue to the model identification stage.

	Variance
Y_t	.1469183
$\nabla_{12}Y_t$	.0687867
$\nabla_{24}Y_t$	.1914766
$\nabla_1\nabla_{12}Y_t$	.100824

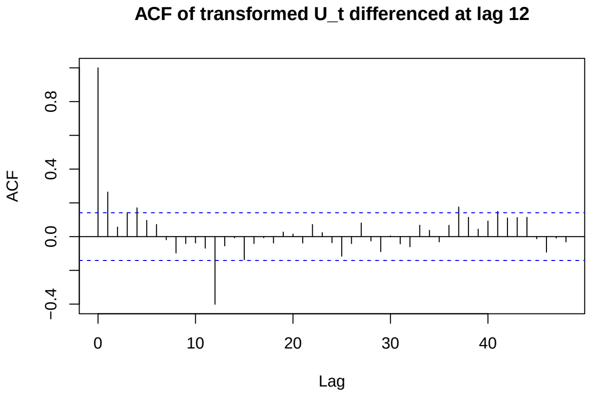


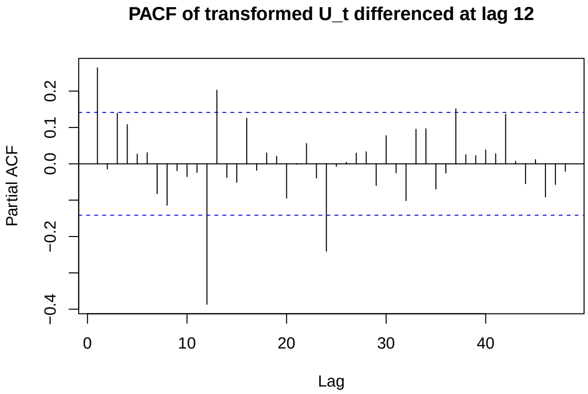
Model Identification

Now identify some possible models to compare. In this section, determine what order each component of the SARIMA$(p, d, q) \times (P, D, Q)_s$ model could be. Since the time series is monthly, $s = 12$. Plot the ACF and PACF of the now stationary time series. $d = 0$ and $D = 1$ since the data was differenced once at lag 12.

The ACF's will help us identify possible q and Q. At lags 1, 4, and 12 the ACF's are large and outside the confidence interval. So consider q=1 or 4, and Q = 12 for the model.

Similarly, the PACF's will help identify possible p and P. Lags 1, 12, and 24 have large PACF's, so consider $p = 1$ and $P = 2$. There is a large lag at 13 as well, but since there is large PACF at lag 1, and lag 12 is one lag away from lag 13, lag 13 does not need to be accounted for. However, due to exponential decay of the seasonal PACF's it is also possible that $P = 0$.





Model Selection

Now test multiple combinations of models, and choose the model with the lowest AICc value to run diagnostics on. First try two pure moving average models, which only have q, Q order components, then one pure auto regressive model which only has p, P order components, before finally trying the mixed models. This is done because finding a model with less coefficents to estimate is the most preferable.

Model	AICc
$(0,0,1) \times (0,1,1)$	-23.680
$(0,0,4) \times (0,1,1)$	-34.671
$(1,0,0) \times (2,1,0)$	-28.419
$(1,0,1) \times (2,1,1)$	-41.532
$(1,0,4) \times (2,1,1)$	-40.065

Choosing the lowest AICc model, SARIMA$(1,0,1) \times (2,1,1)$, notice that the seasonal auto regressive coefficients have zero in their confidence intervals, which means to also test models that do not have these terms by fixing the auto regressed coefficients to be zero and finding the lowest AICc.

Fixed to Zero	AICc
Φ_1	-42.646
Φ_2	-42.919
Φ_1, Φ_2	-44.590

A lower AICc model has been found, so now diagnostic check SARIMA$(1,0,1) \times (0,1,1)$ to see how well it fits as the model for the time series.

Model Diagnostics; unit roots

The model to test is SARIMA$(1,0,1) \times (0,1,1)_{12}$: of the form:

$$(1 - \phi_1 B)(1 - B^{12})X_t = (1 + \theta_1 B)(1 + \Theta_1 B^{12})Z_t$$

Component	Coefficient	Polynomial
AR	$\phi_1 = .9582$	$1 - \phi_1 B$
MA	$\theta_1 = -.7346$	$1 + \theta_1 B$
SMA	$\Theta_1 = -.7534$	$1 + \Theta_1 B^{12}$

Filling in the coefficients estimated from the stationary data for this model:

$$(1 - .9582_{(.0393)} B)(1 - B^{12})X_t = (1 - .7347_{(.0925)} B)(1 - .7534_{(.0640)} B^{12})Z_t$$

where $Z_t \sim WN(0, 0.04225)$

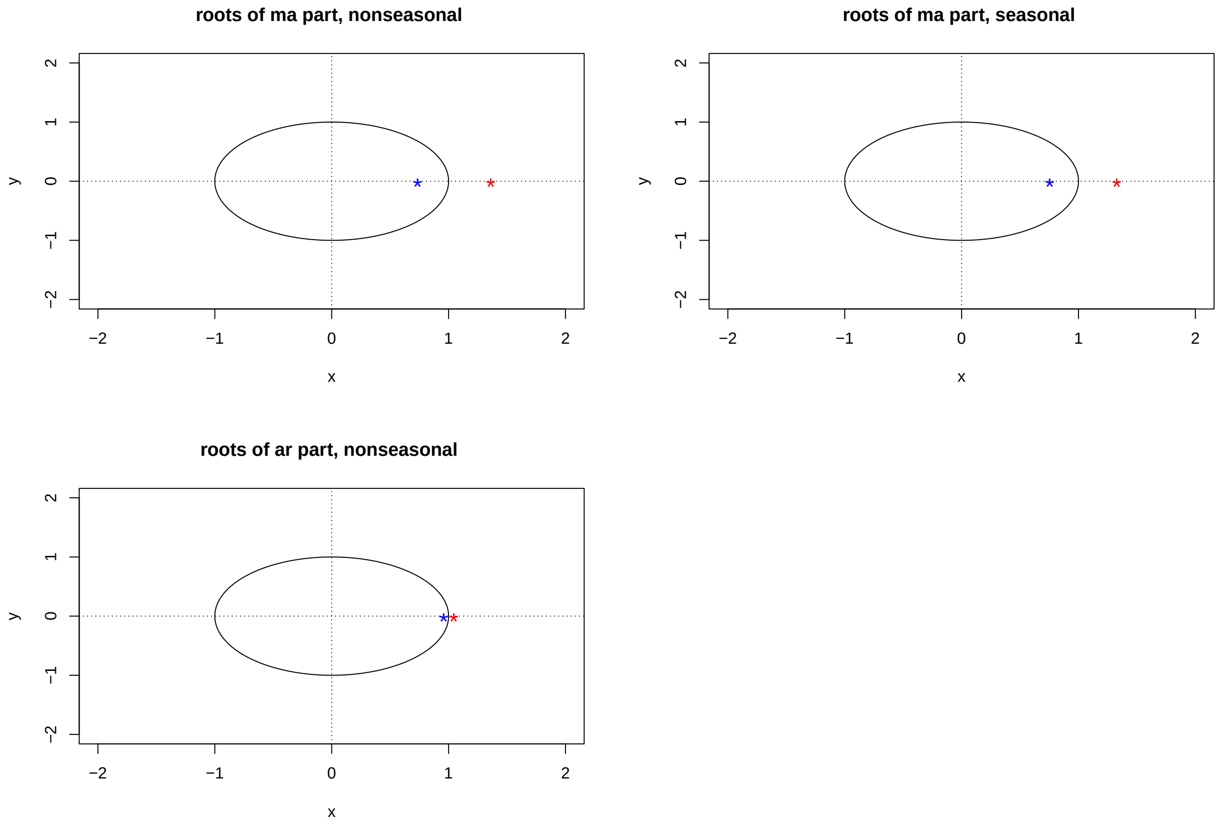
Check that there are no unit roots in the model. This shows stationary, invertibility, and causality and gives a valid model to use. SARIMA models can only model stationary data, invertibilty allows for the past to have less influence on current values, and causality ensures that the observations are future independent.

Since each component of our model is order 1, checking that the corresponding coefficient is less than 1 is sufficient to show that the component is stationary or invertible, depending on which type it already is. Note the characteristic polynomial for the seasonal moving average component is $\Theta(z) = 1 + \Theta_1 z$, so there is still have an order one component even though there is B^{12}.

Since the AR part is defined to be always invertible, having no unit roots means that component is stationary. MA parts are defined to be always stationary, so the SARIMA model is stationary. Since MA parts have no unit roots, they are invertible, and so it follows that the SARIMA model is invertible as well. As for causality, MA is defined to always be casual, and AR(1) is causal when there are no unit roots.

One more helpful result of checking unit roots is that if MA part did have unit roots, there is an overdifference of the data, and if AR had unit roots, there is an underdifference ofthe data.

To visualize, plot the characteristic model roots. Since the red star is outside, the root lies outside of the unit circle.

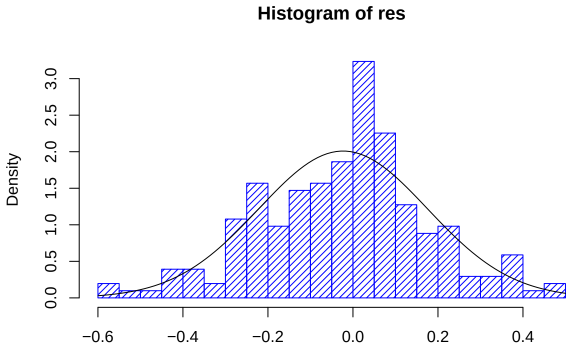


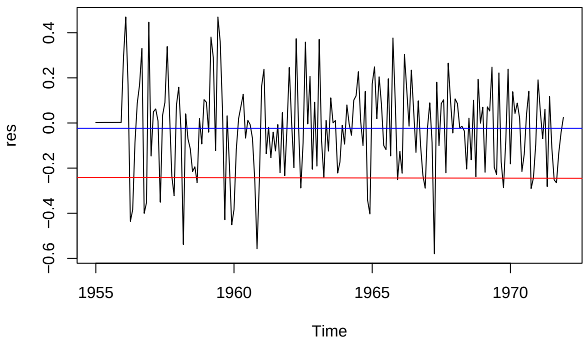
Model Diagnostics; residuals

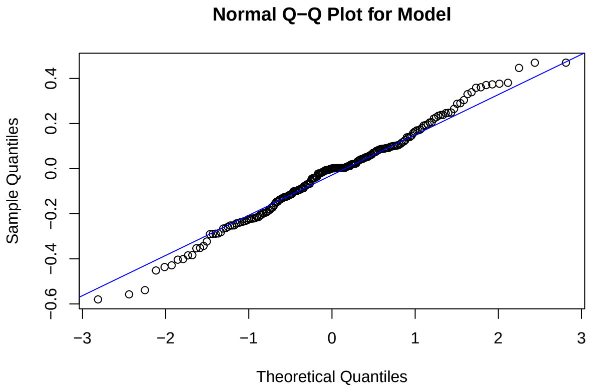
To determine how good the fit of the model being testing is, show residuals are Gaussian White Noise.

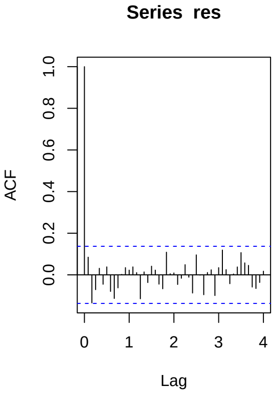
For residuals to be Gaussian White Noise, check the following things:

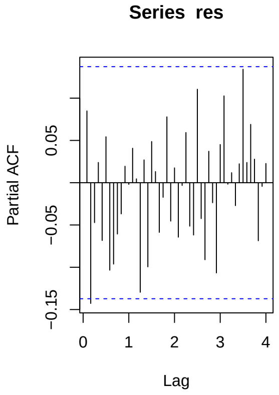
- White Noise: The plot of residuals is stationary.
- Gaussian: Check that the histogram is normally distributed, the Normal Q-Q plot resembles a straight line, and run the Shapiro-Wilk test of normality.
- ACF and PACF are White Noise: Plot the ACF and PACF, should be inside the confidence interval for all lags, h, $h \leq 1$.
- Yule-Walker Estimation: The residuals follow AR(0).
- Portmanteau Tests: Box-Pierce, Ljung-Box, and McLeod-Li tests must pass and fail to reject the white noise hypothesis.











The histogram resembles a Gaussian curve, the points in the Q-Q plot fall on the line, and the Shapiro-Wilk test returns p-value $= .2928 > .05$, so these tests for normality pass. The plot of residuals is white noise, as it has constant mean, no trend, no seasonality, and no change of variance. ACF and PACF of residuals all fall within the confidence intervals, and so they are white noise as well.

Yule-Walker estimation finds that the residuals are AR(2), which shows that $p = 2$ might need to be in the model. Trying SARIMA$(2, 0, 0) \times (2, 1, 0)$ however does not give us a lower AICc, and since all the other tests pass disregard that this test did not pass.

Portmanteau Tests:

By rule of thumb, h is found by $\sqrt{n}$, which in our case is $\sqrt{204}$ which is approximately 14. These tests are run at $\alpha = .05$. Box-Pierce and Ljung-Box degrees of freedom are obtained for χ^2 via $h - p - q$, which in this case would be $14 - 1 - 2 = 11$. This adjustment is needed because there are estimated 3 coefficients in the model. McLeod-Li tests a different statistic from the other two tests and its χ^2 distribution does not have degrees of freedom adjustment.

Test	h	df	p-value	Conclusion
Box-Pierce	14	11	.3291	Fail to reject WN Hypothesis
Ljung-Box	14	11	.299	Fail to reject WN Hypothesis
McLeod-Li	14	14	.6091	Fail to reject WN Hypothesis

Selected Model

The model chosen and passed diagnostics checks is SARIMA$(1,0,1) \times (0,1,1)_{12}$:

$$(1 - \phi_1 B)(1 - B^{12})X_t = (1 + \theta_1 B)(1 + \Theta_1 B^{12})Z_t$$

$$(1 - .9582_{(.0393)}B)(1 - B^{12})X_t = (1 - .7347_{(.0925)}B)(1 - .7534_{(.0640)}B^{12})Z_t$$

where $Z_t \sim WN(0, 0.04225)$

Proceed to forecasting the next year after the training data and comparing it with the test set taken out of the original data. Note that this model is for the transformed and differenced data. After forecasting the prediction with the transformed data, square the forecasts and time series to return to the original data scale.

Forecasting

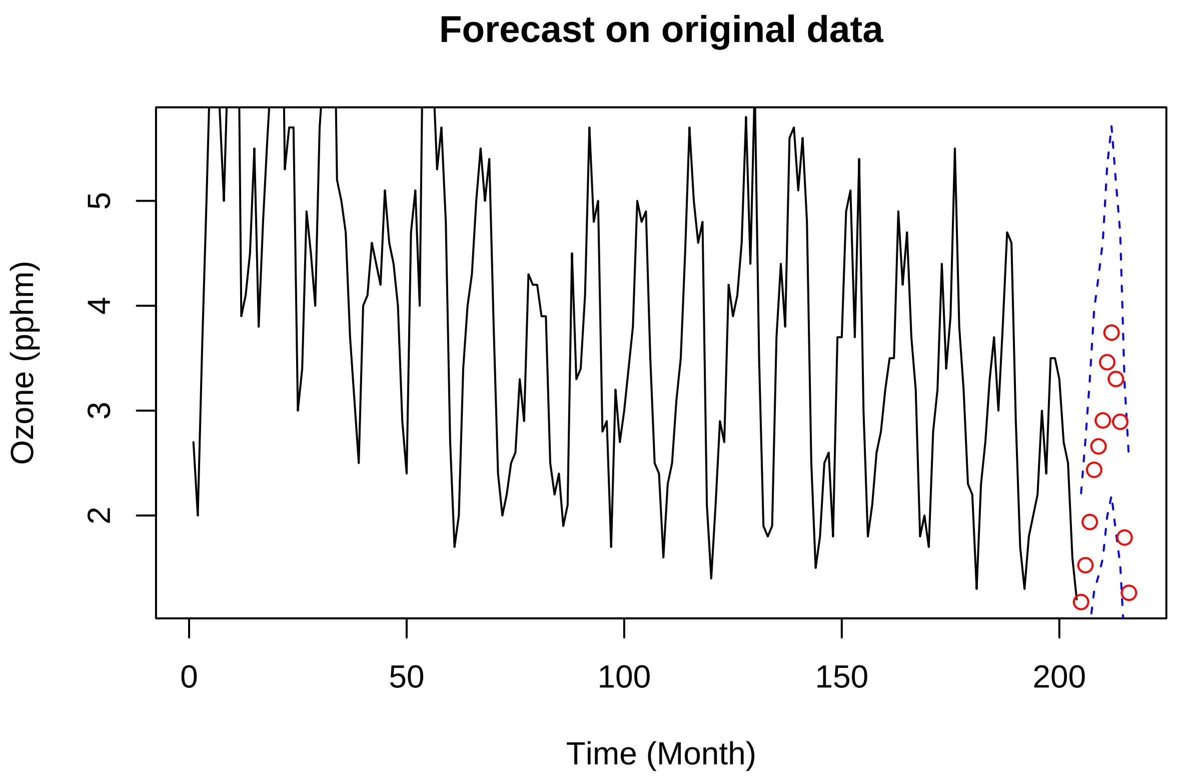


Figure 1: Square the data to untransform it, and show the forecast. Another graph will be produced to zoom in at month 170 and see how good the forecast was.

Zoomed Forecast

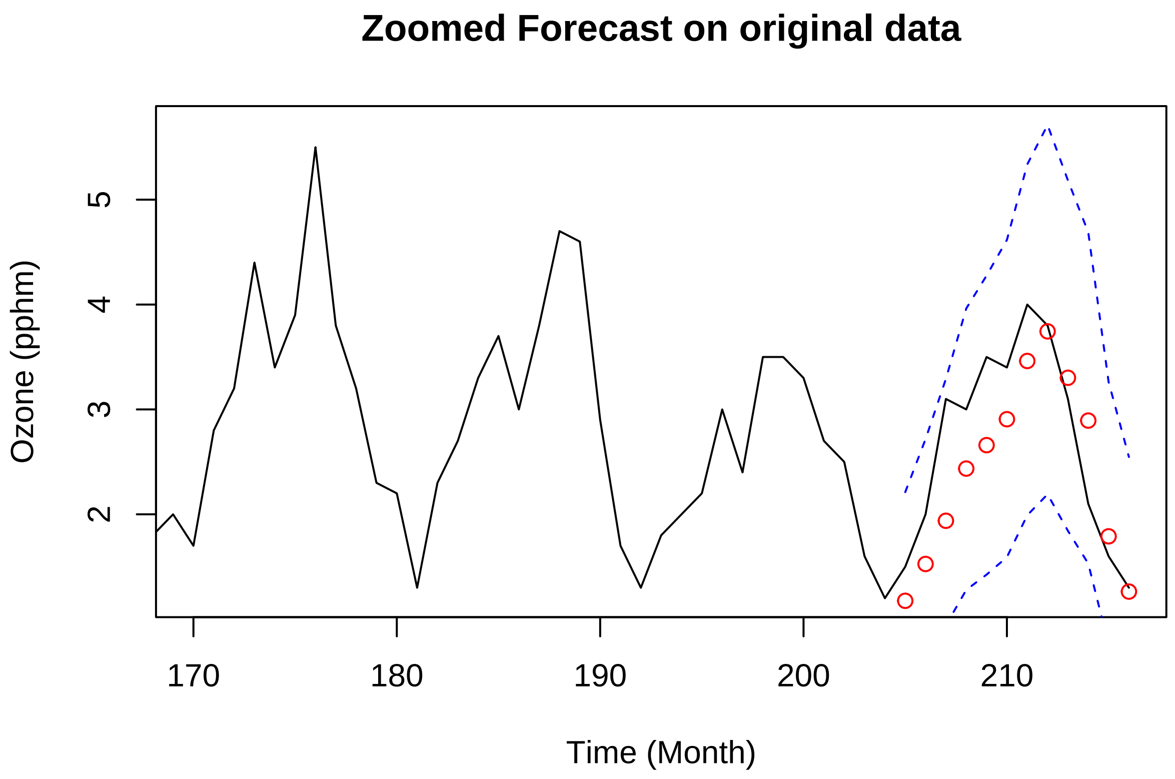


Figure 2: In this graph, the blue dashed lines are the prediction interval, the black line is the original data, and the red points are the forecasted values. Our forecast falls within the prediction interval and follows the curve well, so the model is sufficient.

Conclusion

The final model selected and forecasted on was a SARIMA$(1,0,1)\times(0,1,1)_{12}$,

$$(1-.9582_{(.0393)}B)(1-B^{12})X_t=(1-.7347_{(.0925)}B)(1-.7534_{(.0640)}B^{12})Z_t$$

where $Z_t \sim WN(0, 0.04225)$. This was then squared to represent the data in the original units.

Exploration of this data set was to discover see if the trend and seasonality would be present throughout the years, and what this would mean. Decreasing ozone levels in the atmosphere would be bad for humans due to loss of protection from UV light, and it is evident that ozone levels are slowly decreasing in our model. NASA says that "Skin cancer is the most dramatic result of a too much UV radiation, but there's a lot more too. Photosynthesis in plants is also affected, and that causes problems for the whole food chain." This report on ozone level modeling therefore is important to track and figure out how to keep from destroying the good ozone in the atmosphere with pollutants.

The seasonality was something to note as well, and this pattern is potentially explained by the increase in temperature during the summer. The US Air Purifier article tells us that "Compounds found in vehicle and industrial air pollution, when exposed to sunlight and hot temperatures, can react to form ozone. The combination of more direct sunlight and longer daylight hours creates the summer ozone season." Hitti mentions that increased ozone during the summer months is correlated with an increase of respiratory deaths.

It is also important to mention that this data set is outdated, so the results obtained from the model may not longer hold in the present, but in this report we have complied a methodology to use on modern data.

Acknowledgement is given to Professor Raya Feldman from University of California, Santa Barbara for assistance on this time series analysis report.

References

Andrej-Nikolai Spiess, (2018). Modelling and Analysis of Real-Time PCR Data, (Andrej-Nikolai Spiess, 2018)

Dunbar, Brian. "The Good, the Bad and the Ozone." NASA, NASA, https://www.nasa.gov/missions/earth/f-ozone.html.

Hyndman & Khandakar, JSS (2008). Automatic Time Series Forecasting: the forecast Package for R

Hipel and McLeod (1994), "Ozone concentration, downtown L. A., 1955-1972"

Hitti, Miranda. "Ozone More Deadly in Summer." WebMD, WebMD, 15 Nov. 2004, https://www.webmd.com/children/news/20041115/ozone-more-deadly-in-summer.

Rob Hyndman and Yangzhuoran Yang (2018). tsdl: Time Series Data Library. v0.1.0. https://pkg.yangzhuoranyang.com/tsdl/

"Summer Ozone Season: Why Is Ozone Worse in Hot Weather?" US Air Purifiers, US Air Purifiers, 23 Dec. 2020, https://www.usairpurifiers.com/blog/summer-ozone-season-why-is-ozone-worse-in-hot-weather/.

Venables & Ripley 4th edition, (2002). Functions and datasets to support, "Modern Applied Statistics with S"

Xie, Y. (2021). knitr: A General-Purpose Package for Dynamic Report Generation in R. https://yihui.org/knitr/

Appendix

```r
library(tsdl) # data set library
library(tidyverse) # data wrangling
library(MASS) # for box-cox
library(rgl)  # for AICc
library(qpcR) # for AICc
library(forecast) # for forecasting

# the plot roots function for checking for unit roots
plot.roots <- function(ar.roots=NULL, ma.roots=NULL, size=2, angles=FALSE, special=NULL,
                       sqecial=NULL,my.pch=1,first.col="blue",second.col="red",main=NULL)
{xylims <- c(-size,size)
      omegas <- seq(0,2*pi,pi/500)
      temp <- exp(complex(real=rep(0,length(omegas)),imag=omegas))
      plot(Re(temp),Im(temp),typ="l",xlab="x",ylab="y",xlim=xylims,ylim=xylims,main=main)
      abline(v=0,lty="dotted")
      abline(h=0,lty="dotted")
      if(!is.null(ar.roots))
        {
          points(Re(1/ar.roots),Im(1/ar.roots),col=first.col,pch=my.pch)
          points(Re(ar.roots),Im(ar.roots),col=second.col,pch=my.pch)
        }
      if(!is.null(ma.roots))
        {
          points(Re(1/ma.roots),Im(1/ma.roots),pch="*",cex=1.5,col=first.col)
          points(Re(ma.roots),Im(ma.roots),pch="*",cex=1.5,col=second.col)
        }
      if(angles)
        {
          if(!is.null(ar.roots))
            {
              abline(a=0,b=Im(ar.roots[1])/Re(ar.roots[1]),lty="dotted")
              abline(a=0,b=Im(ar.roots[2])/Re(ar.roots[2]),lty="dotted")
            }
          if(!is.null(ma.roots))
            {
              sapply(1:length(ma.roots),
                     function(j) abline(a=0,b=Im(ma.roots[j])/Re(ma.roots[j]),lty="dotted"))
            }
        }
      if(!is.null(special))
        {
          lines(Re(special),Im(special),lwd=2)
        }
      if(!is.null(sqecial))
        {
          lines(Re(sqecial),Im(sqecial),lwd=2)
        }
        }
# using the tsdl library
tsdl_monthly <- subset(tsdl,12,"Meteorology")
```

```r
ozoneLA <- tsdl_monthly[[7]]

# length of the data
print(length(tsdl_monthly[[7]]))  #216/12=18 years
# subject of the data
print(attr(tsdl_monthly[[7]], "subject"))
# source of the data
print(attr(tsdl_monthly[[7]], "source"))
# description of the data
print(attr(tsdl_monthly[[7]], "description"))


# create training and test set
ozoneLAtrain <- ts(ozoneLA[c(1:204)],start=1955,frequency=12)
ozoneLAtest <- ozoneLA[c(205:216)]
#plot
plot(ozoneLAtrain,main="Ozone concentration, downtown L. A., 1955-1972",
     ylab="Ozone (pphm)")

#histogram
hist(ozoneLAtrain, main = "Ozone concentration, downtown L. A., 1955-1972",
     xlab="Ozone (pphm)") # its right skewed

# acf of U_t
acf<-acf(ozoneLAtrain,lag.max=48,plot=FALSE)
acf$lag <- acf$lag*12
plot(acf, main="ACF of Ozone concentration, downtown L. A.", xlab = "Lag")
# box cox, to deal with the variance
t <- 1:length(ozoneLAtrain)
bcTransform <- boxcox(ozoneLAtrain ~ t, plotit=TRUE)
title(main="Box-Cox Plot")

# try a sqrt since its in the CI of our box cox
ozoneLAtrain.sqrt <- sqrt(ozoneLAtrain)

#decomp
decomp <- decompose(ozoneLAtrain.sqrt)
plot(decomp)

# Plot and compare the original and transformed:
par(mfrow=c(1,2))
ts.plot(ozoneLAtrain, main = "Original data",ylab = expression(U[t]))
ts.plot(ozoneLAtrain.sqrt, main = "Square-Root transformed data", ylab = expression(Y[t]))

# histogram is now approximately normal
par(mfrow=c(1,1))

hist(ozoneLAtrain.sqrt,probability = TRUE)
m<-mean(ozoneLAtrain.sqrt)
std<- sqrt(var(ozoneLAtrain.sqrt))
curve(dnorm(x,m,std), add=TRUE)
var(ozoneLAtrain.sqrt)
```

```r
# Deseasonlize:
y.12 <- diff(ozoneLAtrain.sqrt, 12)
y.12.2 <- diff(y.12, 12)
var(y.12);
var(y.12.2) # variance inc

# Detrend:
y.1.12 <- diff(y.12, 1)
var(y.1.12) # variance inc

# conclude that diff at lag 12 to remove seasonality
# is satisfactory to make a stationary time series
ts.plot(y.12, main="De-seasonlized Time Series",
        ylab=expression(nabla[12]~Y[t]))
abline(h=mean(y.12), lty=2)
# in the end we conclude to work with diff at lag 12

# acf of sqrt diff at lag 12
acf.12<-acf(y.12,lag.max=48,plot=FALSE)
acf.12$lag <- acf.12$lag*12
plot(acf.12, main= "ACF of transformed U_t differenced at lag 12")

# pacf of sqrt diff at lag 12
pacf.12<-pacf(y.12,lag.max=48,plot=FALSE)
pacf.12$lag <- pacf.12$lag*12
plot(pacf.12, main= "PACF of transformed U_t differenced at lag 12")

# so lets find candidates for p and q, note D=12
# via the acf, try q=1 or 4 Q=1
# via pacf, try p=1, P=2
# lag 13 has a huge spike as well though, but since p=1,
#and we have s=12, this may not need to be considered
#model fitting, make sure to check for unit roots

#try the pure models first SMA, Q=1, q= 1 or 4
# Q=1 q=1  (0,0,1)x(0,1,1)
arima(ozoneLAtrain.sqrt, order=c(0,0,1), seasonal = list(order = c(0,1,1),
  period = 12), method="ML")
AICc(arima(ozoneLAtrain.sqrt, order=c(0,0,1), seasonal = list(order = c(0,1,1),
  period = 12), method="ML"))

# Q=1 q=4 (0,0,4)x(0,1,1)
arima(ozoneLAtrain.sqrt, order=c(0,0,4), seasonal = list(order = c(0,1,1),
  period = 12), method="ML")
AICc(arima(ozoneLAtrain.sqrt, order=c(0,0,4), seasonal = list(order = c(0,1,1),
  period = 12), method="ML"))


# try some SAR, P=2, p= 1
# P = 2, p =1  (1,0,0)x(2,1,0)
arima(ozoneLAtrain.sqrt, order=c(1,0,0), seasonal = list(order = c(2,1,0),
  period = 12), method="ML")
AICc(arima(ozoneLAtrain.sqrt, order=c(1,0,0), seasonal = list(order = c(2,1,0),
  period = 12), method="ML"))

# SARIMA, s=12
#p=1
# (1,0,1)x(2,1,1)
arima(ozoneLAtrain.sqrt, order=c(1,0,1), seasonal = list(order = c(2,1,1),
  period = 12), method="ML")
AICc(arima(ozoneLAtrain.sqrt, order=c(1,0,1), seasonal = list(order = c(2,1,1),
  period = 12), method="ML"))
# (1,0,4)X(2,1,1)
arima(ozoneLAtrain.sqrt, order=c(1,0,4), seasonal = list(order = c(2,1,1),
  period = 12), method="ML")
AICc(arima(ozoneLAtrain.sqrt, order=c(1,0,4), seasonal = list(order = c(2,1,1),
  period = 12), method="ML"))
# the model with lowest AICc is SARIMA (1,0,1)x(2,1,1)
arima(ozoneLAtrain.sqrt, order=c(1,0,1), seasonal = list(order = c(2,1,1),
                                                         period = 12), method="ML")

.9566 - 1.96*(.0392) ; .9566  + 1.96*(.0392)

-.7353 - 1.96*(.0939) ; -.7353 + 1.96*(.0939)

-.1417 - 1.96*(.1561) ;-.1417 + 1.96*(.1561)

-.0933 - 1.96*(.1183) ;-.0933 + 1.96*(.1183)

-.6410 - 1.96*(.1481) ; -.6410 + 1.96*(.1481)

# sar1 and sar 2 have 0 in the CI

arima(ozoneLAtrain.sqrt, order=c(1,0,1), seasonal = list(order = c(2,1,1),
  period = 12), fixed=c(NA,NA,0,NA,NA), method="ML")
AICc(arima(ozoneLAtrain.sqrt, order=c(1,0,1), seasonal = list(order = c(2,1,1),
  period = 12), fixed=c(NA,NA,0,NA,NA), method="ML"))

arima(ozoneLAtrain.sqrt, order=c(1,0,1), seasonal = list(order = c(2,1,1),
 period = 12), fixed=c(NA,NA,NA,0,NA), method="ML")
AICc(arima(ozoneLAtrain.sqrt, order=c(1,0,1), seasonal = list(order = c(2,1,1),
  period = 12), fixed=c(NA,NA,NA,0,NA), method="ML"))

arima(ozoneLAtrain.sqrt, order=c(1,0,1), seasonal = list(order = c(2,1,1),
  period = 12), fixed=c(NA,NA,0,0,NA), method="ML")
AICc(arima(ozoneLAtrain.sqrt, order=c(1,0,1), seasonal = list(order = c(2,1,1),
  period = 12), fixed=c(NA,NA,0,0,NA), method="ML"))
# this is the lowest AICc
fit <-arima(ozoneLAtrain.sqrt, order=c(1,0,1), seasonal = list(order = c(2,1,1),
   period = 12), fixed=c(NA,NA,0,0,NA), method="ML")
fit

# variance of white noise
fit$sigma2

plot.roots(NULL,polyroot(c(1, -0.7347)),
           main="roots of ma part, nonseasonal ") # so model is invert

plot.roots(NULL,polyroot(c(1,-.7534)),
           main="roots of ma part, seasonal ") # so model is invert

plot.roots(NULL,polyroot(c(1, -0.9582)),
           main="roots of ar part, nonseasonal ") # model is stationary
# check the residuals model 1

res <- residuals(fit)

# histogram
hist(res,density=20,breaks=20, col="blue", xlab="", prob=TRUE)
m <- mean(res)
std <- sqrt(var(res))
curve(dnorm(x,m,std), add=TRUE)

# plot, we want WN
plot.ts(res)
fitt <- lm(res ~ as.numeric(1:length(res))); abline(fitt, col="red")
abline(h=mean(res), col="blue")

# qqplot
qqnorm(res,main= "Normal Q-Q Plot for Model")
qqline(res,col="blue")

# acf and pacf
par(mfrow=c(1,2))
acf(res, lag.max=48)
pacf(res, lag.max=48)
# looks good, everything is inside
shapiro.test(res) # p > .05, fails to reject H_0 and conclude that res is normal

# Use Yule-Walker estimation: should fit into AR(0)
ar(res, aic = TRUE, order.max = NULL, method = c("yule-walker"))
# it selected order 2, so maybe the p should = 2

# P = 2, p =2  (2,0,0)x(2,1,0)
arima(ozoneLAtrain.sqrt, order=c(2,0,0), seasonal = list(order = c(2,1,0),
                                   period = 12), method="ML")
AICc(arima(ozoneLAtrain.sqrt, order=c(2,0,0), seasonal = list(order = c(2,1,0),
                                   period = 12), method="ML"))
# doesnt provide a lower AICc, proceed with ours and disregard this test failing

# box tests sqrt(204) = appx 14
Box.test(res, lag = 14, type = c("Box-Pierce"), fitdf = 3) # fail to reject WN hyp

Box.test(res, lag = 14, type = c("Ljung-Box"), fitdf = 3) # fail to reject WN hyp

Box.test(res^2, lag = 14, type = c("Ljung-Box"), fitdf = 0) # fail to reject WN hyp
fit <-arima(ozoneLAtrain.sqrt, order=c(1,0,1), seasonal = list(order = c(2,1,1), period = 12),
           fixed=c(NA,NA,0,0,NA), method="ML")
```

```r
# To produce graph with 12 forecasts on transformed data:
pred.tr <- predict(fit, n.ahead = 12)
U.tr <- pred.tr$pred + 1.96*pred.tr$se # upper bound of prediction interval
L.tr <- pred.tr$pred - 1.96*pred.tr$se # lower bound

ts.plot(as.numeric(ozoneLAtrain.sqrt), xlim=c(1,length(ozoneLAtrain.sqrt)+12),
        ylim = c(min(ozoneLAtrain.sqrt), max(U.tr)),
        main="Forecast on transformed data", ylab="Ozone (pphm)")
points((length(ozoneLAtrain.sqrt)+1):(length(ozoneLAtrain.sqrt)+12), pred.tr$pred, col="red")
lines((length(ozoneLAtrain.sqrt)+1):(length(ozoneLAtrain.sqrt)+12),U.tr, col="blue", lty="dashed")
lines((length(ozoneLAtrain.sqrt)+1):(length(ozoneLAtrain.sqrt)+12),L.tr, col="blue", lty="dashed")
#To produce graph with forecasts on original data:
pred.orig <- (pred.tr$pred)^2
U <- (U.tr)^2
L <- (L.tr)^2
ts.plot(as.numeric(ozoneLAtrain), xlim=c(1,length(ozoneLAtrain)+12),
        ylim = c(min(ozoneLAtrain),max(U)),
        main="Forecast on original data",xlab="Time (Month)", ylab="Ozone (pphm)")
lines((length(ozoneLAtrain)+1):(length(ozoneLAtrain)+12),U, col="blue", lty="dashed")
lines((length(ozoneLAtrain)+1):(length(ozoneLAtrain)+12),L, col="blue", lty="dashed")
points((length(ozoneLAtrain)+1):(length(ozoneLAtrain)+12), pred.orig, col="red")


#To zoom the graph, starting from entry 170
ts.plot(as.numeric(ozoneLAtrain), xlim=c(170,length(ozoneLAtrain)+12),
        ylim = c(min(ozoneLAtrain),max(U)),
        main="Zoomed Forecast on original data",ylab="Ozone (pphm)")
lines((length(ozoneLAtrain)+1):(length(ozoneLAtrain)+12),U, col="blue", lty="dashed")
lines((length(ozoneLAtrain)+1):(length(ozoneLAtrain)+12),L, col="blue", lty="dashed")
points((length(ozoneLAtrain)+1):(length(ozoneLAtrain)+12), pred.orig, col="red")
#To plot zoomed forecasts and true values (in ozoneLA):
ts.plot(as.numeric(ozoneLA), xlim=c(170,length(ozoneLAtrain)+12),
        ylim = c(min(ozoneLAtrain),max(U)),
        main="Zoomed Forecast on original data",xlab="Time (Month)",ylab="Ozone (pphm)")
lines((length(ozoneLAtrain)+1):(length(ozoneLAtrain)+12),U, col="blue", lty="dashed")
lines((length(ozoneLAtrain)+1):(length(ozoneLAtrain)+12),L, col="blue", lty="dashed")
points((length(ozoneLAtrain)+1):(length(ozoneLAtrain)+12), pred.orig, col="red")
```